Exam 2

March 3, 2016

Create a markdown file in RStudio, and call it `YourLastName_Exam2`. Answer all of the exam questions in this file, with your r code for each problem inserted in a separate chunk. When you are finished, knit your file to `YourLastName_Exam2.html`. Make sure your markdown file compiles properly, copy both files to a flash drive, and bring them to the front to turn them in before you leave. Also turn in the hard copy of the exam with your name written at the top.

Metapopulations are local populations that are connected through immigration and emmigration. The variable f is used to indicate the proportion of patches in a landscape that are occupied by local populations. The classic island-mainland model for metapopulations is:

$$\frac{df}{dt} = i(1-f) - ef$$

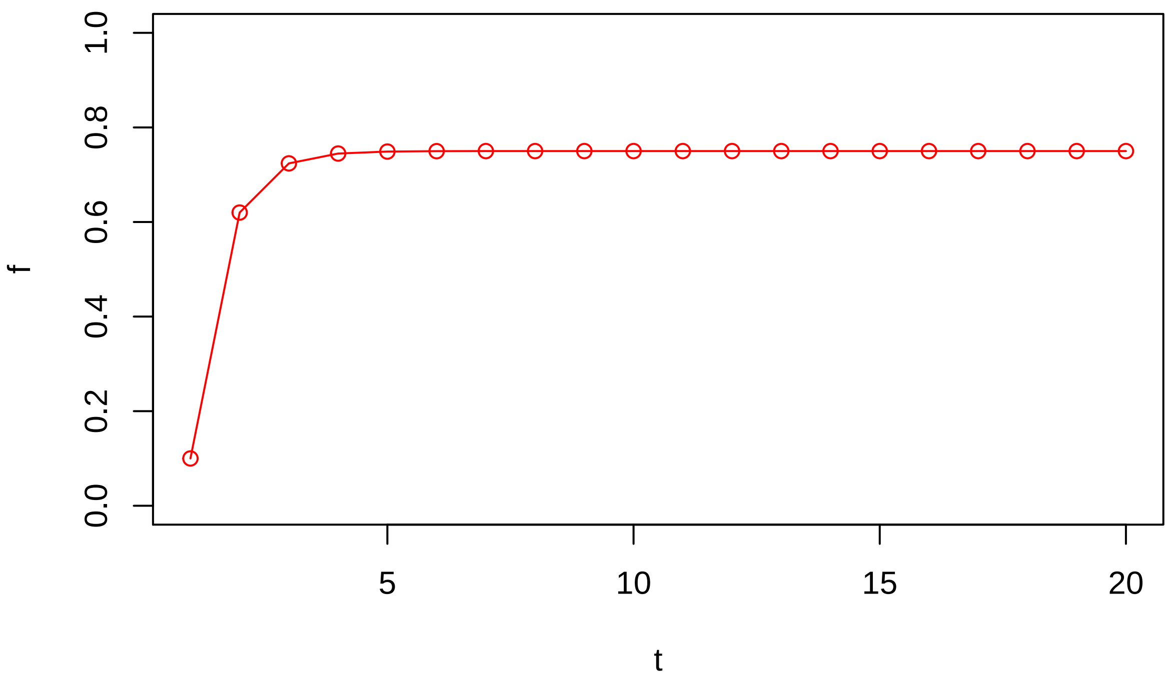
where i and e are parameters that represent the probability of local colonization and local extinction.

1. Write a function `Model1` that takes as input the following parameters:

- `Time` = the number of time steps (set to a default value of 20)
- `f_0` = the initial vaue for f (set to a default value of 0.1)
- `i` = the immigration parameter (set to a defaut value of 0.6)
- `e` = the extinction parameter (set to a default value of 0.2)

Your function should return a vector of length `Time` that contains the values of f at each time step (Be careful! If you are using i as the name of one of your parameters, you cannot also use i as a counter variable in a `for` loop!). It should also generate a simple plot of f versus t. In the plot, set the y-limits to be 0 and 1 (because f is a proportion, it will always be between these boundaries). Run the function with its default values. **(15 points)**

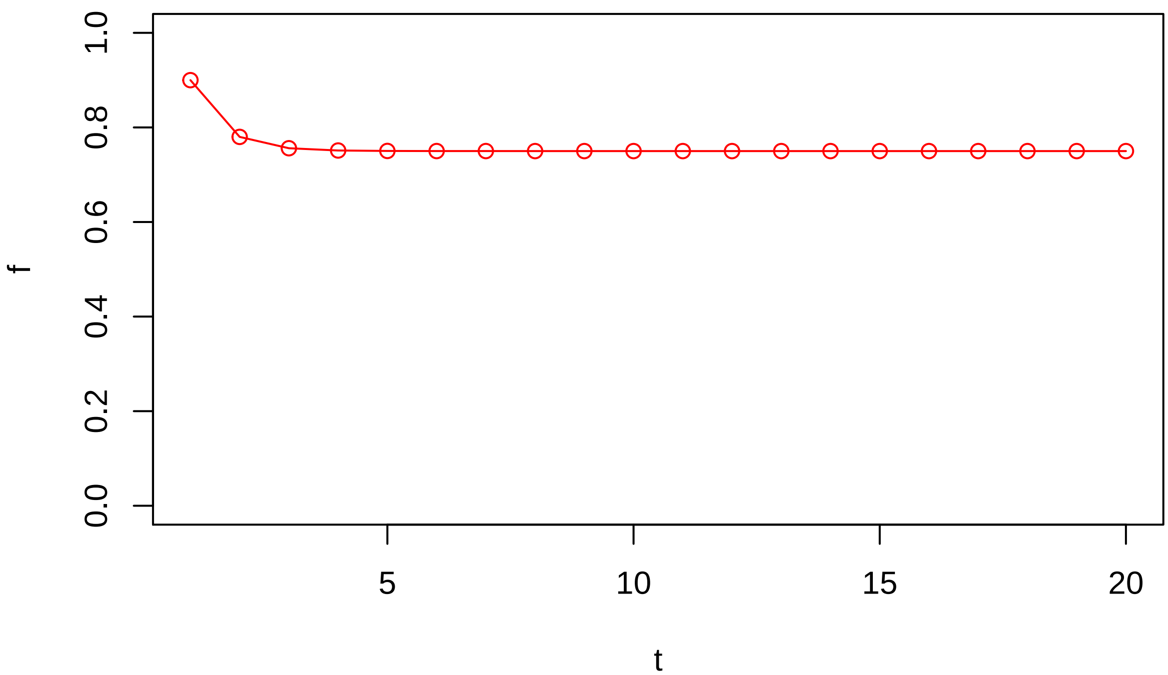
```
# Model1 generates a simple island-mainland metapopulation model
Model1 <- function(Time=20,f_0=0.1,i=0.6,e=0.2){
  myVec <- rep(0,Time)
  myVec[1] <- f_0
  for (j in 2:Time){
    myVec[j] <- myVec[j-1] + i*(1 - myVec[j-1]) - e*myVec[j-1]
  }
  plot(x=1:Time,y=myVec,
              type="o",
              xlab="t", ylab="f",
              col="red",
              ylim=c(0,1))


}

Model1()
```



2. Using `Model1`, illustrate different input values for *f_0*, *i* and *e* (remember, *i* and *e* are probabilities, and *f_0* is a proportion, so all three must be real numbers between 0.0 and 1.0). Based on these values, describe in your markdown file how the variable *f* changes through time and how it is affected by *f_0*, *i* and *e*. **(15 points)**

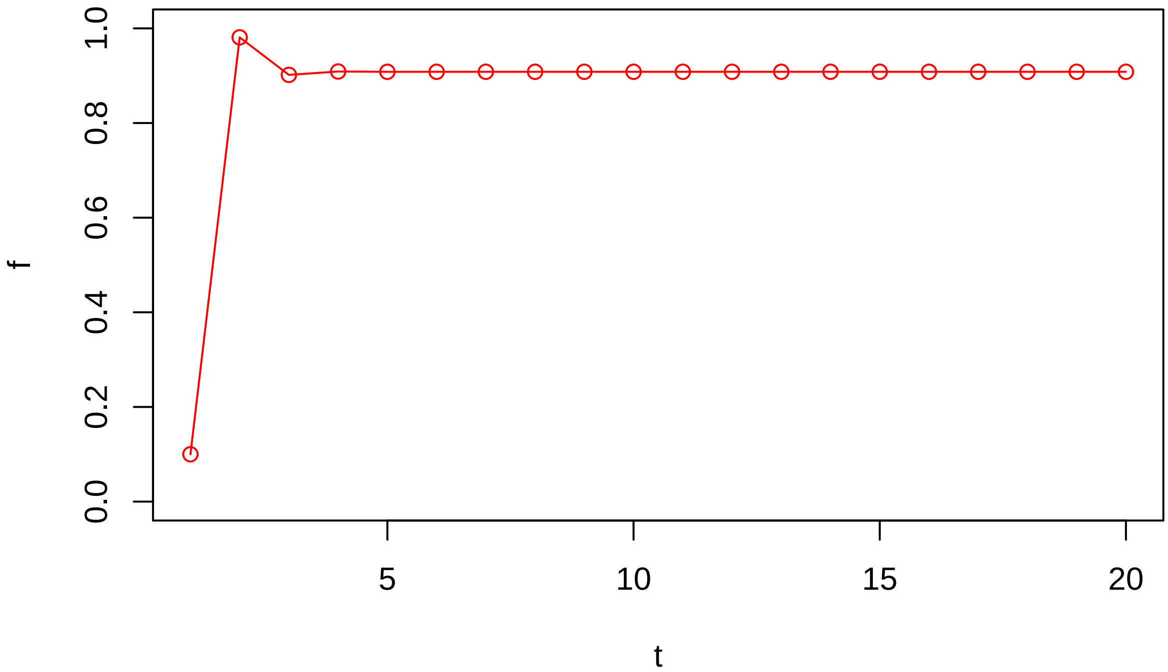
   If we change the starting value, the value of *f* quickly goes to the same equilibrium:

```
Model1(f_0=0.9)
```



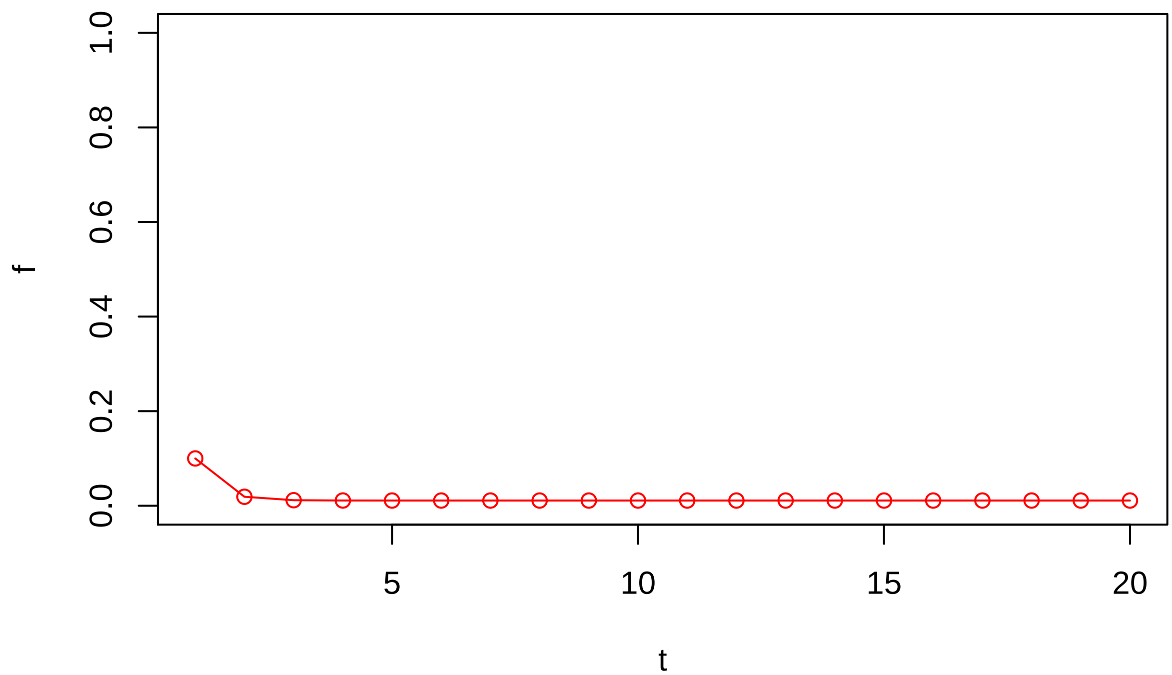
If we make i relatively large, then the final value of f is also relatively large:

```
Model1(i=0.99,e=0.1)
```



If we make i relatively small, then the final value of f is also relatively small:

```
Model1(i=0.01,e=0.9)
```

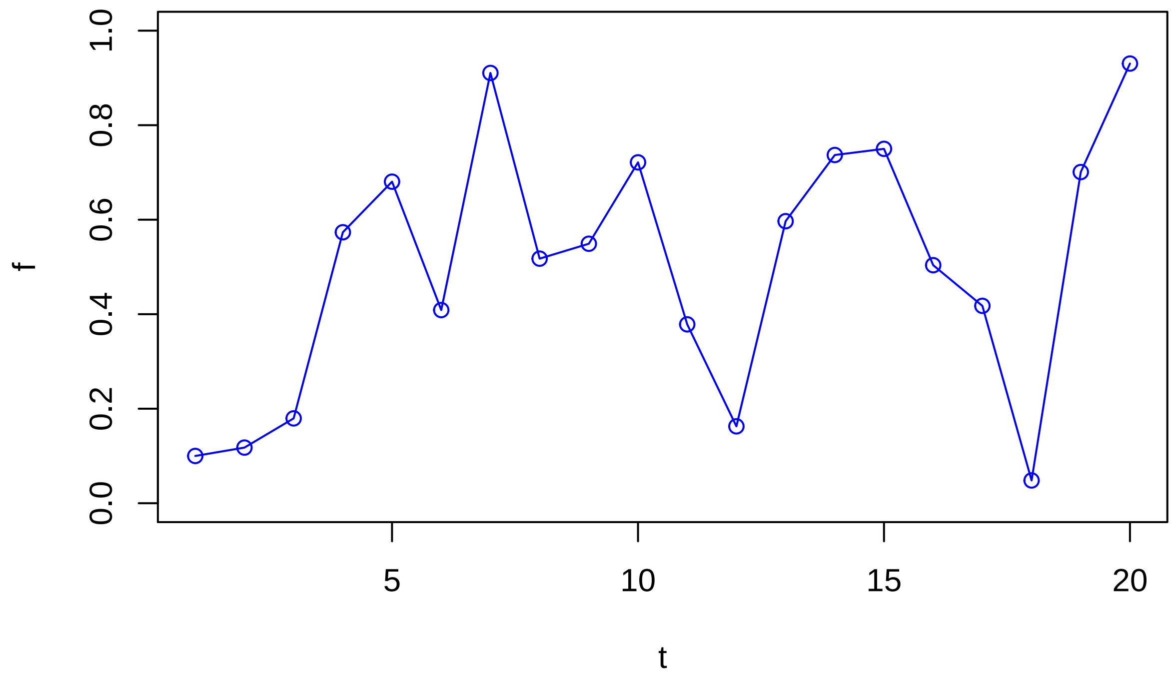


So, there is an equilibrium for this equation that does not depend on the initial value. The large i or the smaller e, the closer the equilibrium is to 1.0.

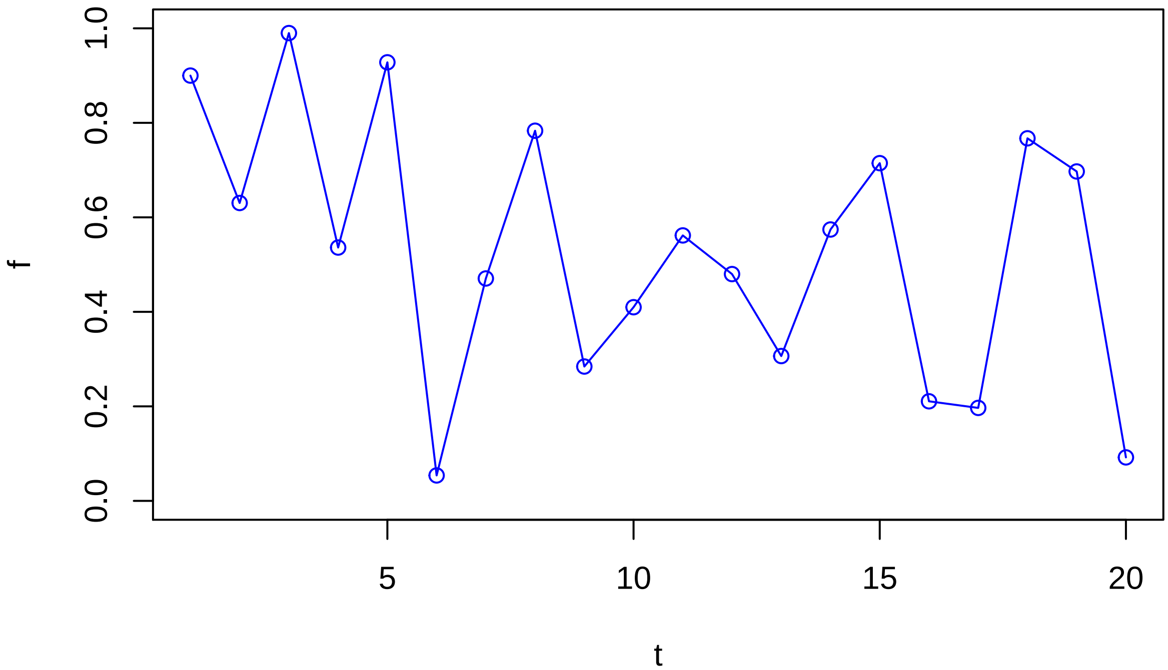
3. Create `Model2` from a copy of `Model1`. Keep all of the inputs the same. But instead of assigning a constant for the parameters i and e, allow these parameters to take on a different random value (use `runif`, which will keep the values between 0 and 1) at each time step. As part of the output of `Model2`, generate a simple plot of f versus t. How does the output from the deterministic `Model1` differ from the stochastic `Model2`? **(20 points)**

```
# Model2 generates a simple stochastic island-mainland metapopulation model
Model2 <- function(Time=20,f_0=0.1,i=0.6,e=0.2){
  myVec <- rep(0,Time)
  myVec[1] <- f_0
  i=runif(Time)
  e=runif(Time)
  for (j in 2:Time){
    myVec[j] <- myVec[j-1] + i[j-1]*(1 - myVec[j-1]) - e[j-1]*myVec[j-1]
  }
  plot(x=1:Time,y=myVec,
              type="o",
              xlab="t", ylab="f",
              col="blue",
              ylim=c(0,1))
```

```
}

Model2()
```




```
Model2(f_0=0.9,i=0.9,e=0.1)
```



With this model, it doesn't seem to matter what the initial *f_0* is, and it doesn't matter what e and i are. The value of f bounces around between 0 and 1 and does not reach a constant equilibrium.